Spike-Based Convolutional Network for real-time processing

J. A. Pérez-Carrasco[1, 2], C. Serrano[2], B. Acha[2], T. Serrano-Gotarredona[1], B. Linares-Barranco[1]
[1]Instituto de Microelectrónica de Sevilla (IMSE-CNM-CSIC)
E-mail: {jcarrasco, terese, bernabe}@imse-cnm.csic.es
[2]Dpto. Teoría de la Señal, ETSIT, Universidad de Sevilla
E-mail: {cserrano, bacha}@us.es

***Abstract*—In this paper we propose the first bio-inspired six-layer convolutional network (ConvNet) non-frame based that can be implemented with already physically available spike-based electronic devices. The system was designed to recognize people in three different positions: standing, lying or up-side-down. The inputs were spikes obtained with a motion retina chip. We provide simulation results showing recognition delays of 16 milliseconds from stimulus onset (time-to-first spike) with a recognition rate of 94%. The weight sharing property in ConvNets and the use of AER protocol allow a great reduction in the number of both trainable parameters and connections (only 748 trainable parameters and 123 connections in our AER system (out of 506998 connections that would be required in a frame-based implementation).**

Keywords-convolutional networks, AER, backpropagation

I. INTRODUCTION

Architectures based on frames have difficulties to solve the nowadays even more computationally demanding operations. An alternative could be to mimic brain behavior. Brains do not use the frame concept to process [1][2]. In the retina, spikes (also called events) are sent to the cortex when the retina pixels reach a threshold. Very active pixels will send more spikes than less active pixels. These spikes are processed and communicated from one layer to the following without waiting for the reception of the whole frame ("frame time") before starting computations in each layer. One big problem encountered to implement bio-inspired (vision) processing systems is to overcome the massive interconnections among the neural layers existing in the human vision processing system. The Address Event Representation (AER) [3] approach, where pixel intensity is coded directly as pixel event frequency, is a possible solution. In AER, each time a pixel generates a spike, its x,y address is written on a shared bus. AER has been used in many applications [4][5][6] and several AER fully-programmable-kernel convolution chips [7] allowing the design of several-layers convolution-based systems have been reported. In AER-based convolution chips, the convolution sum is implemented by adding a projection field (the convolution map) in a pixel array around the address coded by each one of the incoming spikes (or events). The results are the same in frame-based and AER-based implementations. However, in AER, only active pixels at the input produce spikes. Whenever an output pixel in the pixel array exceeds a fixed threshold level, it generates an output event, and the firing pixel is reset. This concept allows the assembly of multi-layer systems where events are generated, transmitted, and processed immediately, without waiting for any frame timing constraints. However, at present, complex AER-based applications and large scale hardware systems have not been reported yet. Probably the largest and complex AER system reported so far is the CAVIAR system [8], which uses four custom made AER chips (motion retina, convolution chip, winner-take-all chip, and learning chip) plus a set of FPGA based AER interfacing and mapping modules.

In the present work we propose the first bio-inspired six-layer convolutional network (ConvNet) non-frame based that can be implemented with already physically available AER-based electronic devices [5][7].

II. METHODOLOGY

Convolutional Neural Networks (ConvNets) [9][10] implement powerful applications in image and video processing [2][10]. ConvNets have a graceful scaling capability and they combine local receptive fields, shared weights and spatial subsampling to ensure some degree of shift, scale, and distortion invariance. In this work we propose a six-layer ConvNet similar to the Convolutional Network LeNet-5 implemented by Y. LeCun [10]. However our system has been implemented directly in AER to detect people in vertical, up-side-down and laying positions from real input data obtained with a physically available temporal contrast (motion) 128x128 AER-based retina [4].

A. Frame-based Convolutional Network

First, the system was implemented and trained using a frame-based architecture to get all the values of the weights and biases present in the system. The frame-based version of our AER six-layer ConvNet is shown in Fig. 1. The system receives as inputs 32x32 images obtained after collecting spikes from the electronic AER retina during 30ms. The output of each one of the six layers in the system is a set of output images or planes called feature maps. A feature map in one layer is only connected to a feature map in the following layer. There are no connections between neurons inside one layer. A unit $x^q(i,j)$ (also called pixel or neuron) located at position *(i,j)* inside a feature map x^q (of size *KxL*) will have a value that follows the expression:

$$x^q(i,j) = A\tanh(S*((\sum_{p\in P}\sum_{m\in M}\sum_{n\in N} y^p(m,n)*W^{q,p}(i-m,j-n))+b^q)). \quad (1)$$

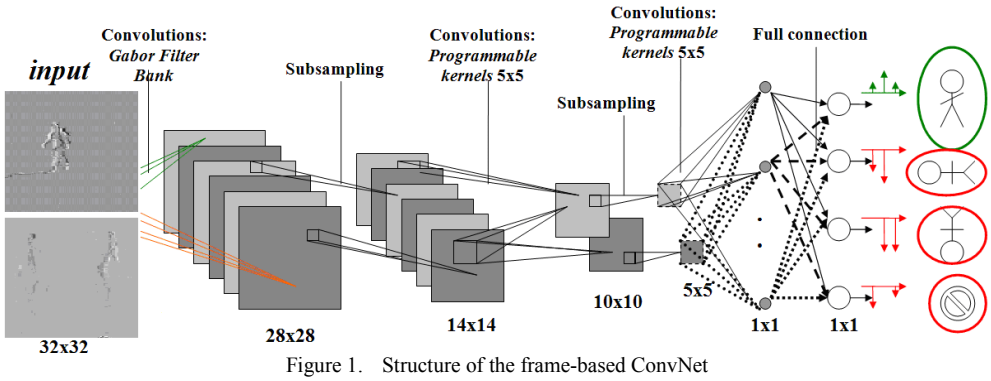

Figure 1. Structure of the frame-based ConvNet

Where y^p is the input feature map p (of a previous layer) of size MxN, $W^{q,p}$ is the map of weight values connecting output feature map x^q with input feature map y^p and b^q is the bias corresponding to the output units located in feature map x^q. A and S are constants.

Similarly to LeNet-5 [10] the proposed system has six layers. As in [10], the inputs to the system are 32x32 images. The first layer *(C1)* of our system is a trainable filter bank with 6 filters and six 28x28 output feature maps, the second layer *(S2)* is a subsampling block with six 14x14 output feature maps, the third layer *(C3)* is a trainable 5x5 kernels filter bank with two 10x10 output feature maps, the fourth layer *(S4)* is a subsampling block again with two 5x5 output feature maps, the fifth layer *(C5)* is a trainable 5x5 kernel filter bank with eight 1x1 output feature maps and sixth layer *(C6)* is a fully connected trainable perceptron with four output units. However, we have substituted the trainable first layer by a bank of 10x10 Gabor filters with two scales and three orientations because this way we do not need to train the first layer thus avoiding this way 606 trainable parameters and because a bank of Gabor filtering is often the first stage of visual processing in many systems and in the human brain [1][2]. In addition, Gabor filters are selective to different scales and orientations and they remove noise due to sparse spikes produced by the retina. The whole system has only 748 trainable parameters, which have been computed using backpropagation [10] and 506998 connections.

B. *AER-based Convolutional Network*

In Fig. 2 the non-frame-based AER-based system is shown. The input is a flow of events captured with an AER motion sensing retina. Each input event is replicated to six output channels using a splitter module [5]. The bank of Gabor filters in first layer is implemented using a set of six AER convolution chips [7] programmed with these Gabor filters as kernels. The output spikes (coding each one of the 28x28 address space of each output image) are sent to subsampling modules. These modules can be easily implemented as follows: address of each input spike is modified so that address *(i,j)*, for *i,j=1,…,28*, turns to *(k,l)*, for *k,l=1,…,14*. The output of each subsampling module is sent to a splitter to replicate the output in two channels and

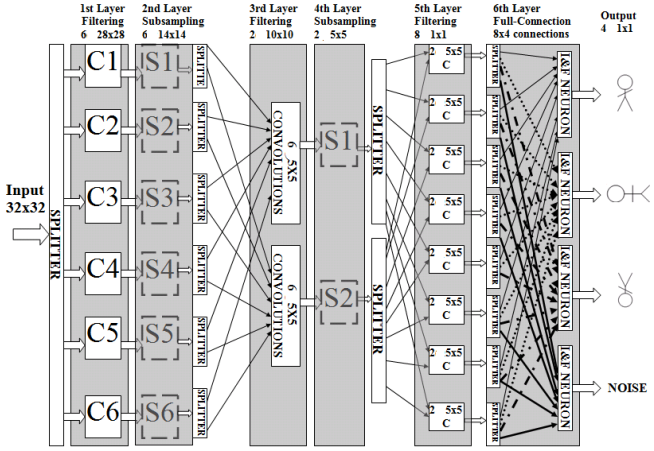

Figure 2. AER-based ConvNet

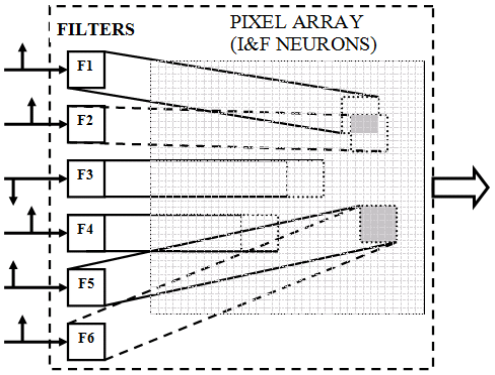

Figure 3. Convolution structure in third layer

each channel is connected to one input of the two convolution structures with six input ports available in third layer. One of these convolution structures is shown in Fig. 3. In this structure, each time a spike is received, a convolution map (projection field) is added around the input spike address in the pixel array (output feature map). Each time a neuron (unit or pixel) in the feature map reaches a threshold and the time since the last output spike (*Toutput*) is higher than an established time value (*Trefractory*), a new output spike is fired to the following layers coding the neuron address and the neuron is reset to the bias value in the feature

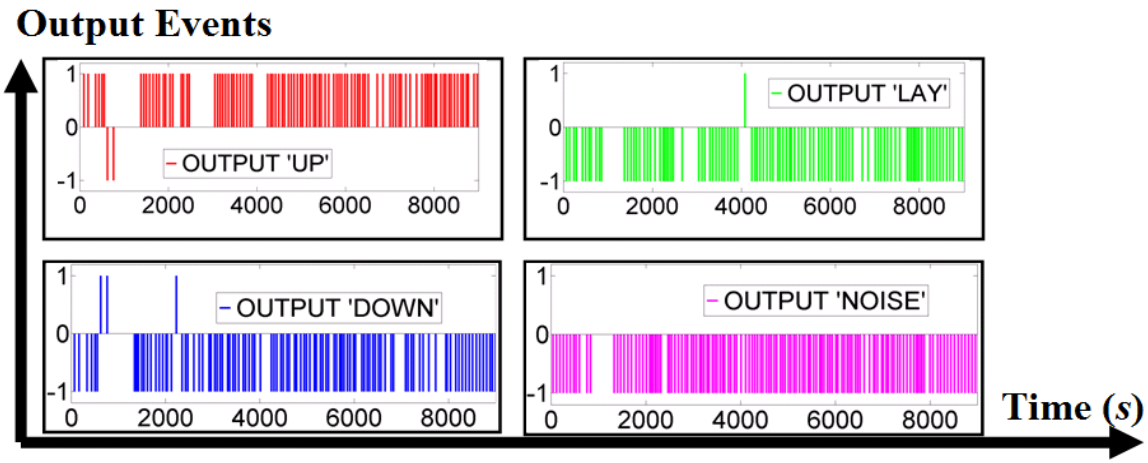

Figure 4. Output events when input is UP

map. Using this idea of limiting the neurons maximum spiking rate with these refractory periods [4] to allow neurons to fire, we solve an important problem found in AER which is the computation of non-linearities, since with already available hardware AER devices it is not easy to implement sigmoid functions. Refractory times were used in third and fifth layers to emulate the corresponding sigmoid functions implemented in the same layers of the frame-based implementation.

The number of events fired by a neuron at position *(i,j)* in the feature map x^q is computed as

$$eout^q(i,j) = \frac{\sum_{p\in P}\left(\sum_{h\in Nevs}\sum_{m\in M}\sum_{n\in N}(ein_h^p(m,n) * W_q^p(i-m,j-n))\right)}{Threshold^q}. \quad (2)$$

Where ein_h^p is the event h coming to input port p coding the address *(m,n)*, W_q^p is the convolution map connecting feature map x^q with input feature map y^p and $Threshold^q$ is the threshold selected for output feature map x^q. The number of events fired by a neuron in layers C3 and C5 is limited to that allowed by the corresponding refractory periods in those layers.

Layer S4 implements subsampling again. Output spikes from layer S4 are connected to neurons in layer C5 in the same way as in layer C3. Each spike produced in layer C5 is replicated in four different outputs which are fully connected to the four output neurons in layer F6. Neurons at layer F6 will fire positive or negative events indicating that the input has been or not categorized as the class coded by the firing neuron.

III. RESULTS

We have simulated the proposed system with our validated AER C++ Simulator Tool [6] as the large number of filters needed is not available electronically yet. However, the system was simulated with real input stimuli from a retina chip and the performance figures from already physically available AER hardware [7][8].

The system was first trained and tested using the frame-based version depicted in Fig. 1. First, we used the events captured during 8s with the electronic retina to create 32x32 images by collecting spikes in intervals of 30ms. This way we created 262 images of people walking. We rotated these images to create the corresponding images in horizontal and up-side-down positions. Finally, to add some distracters we used 262 noise images of some objects moving obtained with the retina. Thus, we generated a database composed of 1048 images representing a total of four different categories. Using 600 of these images to train the system and the rest 448 to test it, we obtained a 98% recognition rate with the training set and 93.2% with the testing set.

All the weights, filters and biases computed in the training stage with this frame-based version of our system were then used in the AER frame-free system. To compute the refractory periods to be used in layers C3 and C5, we established relations between the values that produce the saturation of the sigmoid functions in the frame-based implementation and the number of events fired by a neuron in the same layers in the AER-based implementation. This way we obtained the values 0.5ms and 23ms for refractory periods in layers C3 and C5 respectively.

Finally, the AER system was tested with three new flows of spikes of visual information. The first flow (corresponding to up position) was directly obtained using the AER motion sensing retina capturing spikes during 10s. Several people appear in the recording at different times (the images obtained after reconstruct the spikes captured by the retina are similar to those appearing in Fig. 1). Then, the flow of spikes (corresponding to up position) was rotated 90 and 180 degrees to create the corresponding flows for lying and up-side-down positions. As each flow had 102572 spikes, this supposed an approximate firing input of 10keps (kilo-events per second). The system was tested with these three different flows as input. In Fig. 4 we show the four output channels when the input testing flow corresponds to the up position. Positive events in a particular output channel indicate that the system recognizes input events as belonging to the

category represented by that output channel. Negative events indicate the opposite. We have considered the recognition rate to be the ratio between the positive events due only to the output channel in which we are interested and the total positive output events. Computing the results for the three different input flows a total performance accuracy of 94% was obtained. The system misclassified mainly up position (classifying it as the up-side-down) when there were transitions between people. The minimum time-to-first spike (time to get the first correct output spike since the input stimulus was presented) indicating the system delay was of 16ms. This means that only over 160 input spikes were needed, which corresponded approximately to an average of 3 spikes per pixel. The maximum firing rate in each output channel leaded to minimum delays between spikes in the order of 15ms. These delays are very low, but even so they are not determined by computation process within the convnet, in the order of microseconds [7], but by the reduced input firing frequency provided by the retina.

IV. CONCLUSIONS

The AER system developed constitutes the first step in the design of learned frame-free bio-inspired systems. Moreover, the design of such systems is becoming a reality as ConvNets show good up scaling behavior and are susceptible of being implemented efficiently with new nano scale hybrid CMOS/nonCMOS technologies [11]. The weight sharing property in ConvNets and the use of AER protocol allow a great reduction in the number of both trainable parameters and connections. Our AER system has only 748 trainable parameters and 123 connections (out of 506998 connections in the frame-based version).

The AER system is able to work in real time providing very quick output flows. This real-time working possibility is due to the availability of AER convolution chips with delays between input and output flows of events of the order of microseconds, making both flows in practice simultaneous. In a frame-based system the high number of convolutions (38) would suppose a bottleneck problem. Furthermore, the combination of Gabor filters in the first layer together with a motion sensing retina allows the removal of static objects and noise together with the selection of certain scales at different orientations. This two-step preprocessing stage also supposes a high computation load in most of frame-based systems.

ACKNOWLEDGMENT

This work was supported by Spanish grants TEC2006-11730-C03-01 (SAMANTA2), and Andalusian grant P06TIC01417 (Brain System). JAPC was supported by an Andalusian scholarship.

REFERENCES

[1] G. M. Shepherd, The Synaptic Organization of the Brain. Oxford Univ. Press, 1990.

[2] T. Serre, et al., "Robust Bobject Recognition with Cortes-Like Mechanisms", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 29(3), March 2007, pp. 411-426.

[3] K. Boahen, "Point-to-Point Connectivity Between Neuromorphic Chips Using Address Events", IEEE Trans. Circ. Sys. II, vol. 47(5), pp. 416-434, 2000.

[4] P. Lichtsteiner, C. Posch, and T. Delbrück, "A 128x128 120dB Latency Asynchronous Temporal Contrast Vision Sensor", IEEE J. Solid-State Circuits, vol. 43(2), February 2008, pp. 566-576.

[5] R. Serrano-Gotarredona, et al., "AER Building Blocks for Multi-Layers Multi-Chips Neuromorphic Vision Systems", Advances in Neural Information Processing Systems, 18, Y. Weiss and B. Schölkopf and J. Platt (Eds.), (NIPS'06), MIT Press, Cambridge, MA, pp. 1217-1224, 2006.

[6] J.A. Pérez-Carrasco, C.Serrano, B. Acha, T. Serrano-Gotarredona, B. Linares-Barranco, "Spike-Based Simulation and Processing. Lecture Notes in Computer Science", vol. 5807, pp. 640-651, 2009.

[7] R. Serrano-Gotarredona, et al., "On Real-Time AER 2D Convolutions Hardware for Neuromorphic Spike Based Cortical Processing", IEEE Trans. On Neural Networks, vol. 19(7), July 2008, pp. 1196-1219.

[8] R. Serrano-Gotarredona, et al., "CAVIAR: A 45k-Neuron, 5M-Synapse, 12G-connects/sec AER Hardware Sensory-Processing-Learning-Actuating System for High Speed Visual Object Recognition and Tracking", IEEE Trans. On Neural Networks, vol. 20(9), September 2009, pp. 1417-1438.

[9] K. Fukushima and N. Wake, "Handwritten Alphanumeric Character Recognition by the Neocognitron", IEEE Trans. Neural Networks, vol. 2(3), May 1991, pp. 355-365.

[10] Y. LeCun, et al., "Gradient-Based Learning Applied to Document Recognition", Proceedings of the IEEE, vol. 86(11), November 1998, pp. 2278-2324.

[11] D. B. Strukov et al., "CMOL FPGA: a configurable architecture for hybrid digital circuits with two-terminal nanodevices", Nanotechnology, vol. 16, pp. 888-900, 2005.